



Validation of a Robust Neural Real-Time Voltage Estimator for Active Distribution Grids on Field Data

Pertl, Michael; Douglass, Philip James; Heussen, Kai; Kok, Koen

Published in:
Electric Power Systems Research

Link to article, DOI:
[10.1016/j.epsr.2017.08.016](https://doi.org/10.1016/j.epsr.2017.08.016)

Publication date:
2018

Document Version
Peer reviewed version

[Link back to DTU Orbit](#)

Citation (APA):
Pertl, M., Douglass, P. J., Heussen, K., & Kok, K. (2018). Validation of a Robust Neural Real-Time Voltage Estimator for Active Distribution Grids on Field Data. *Electric Power Systems Research*, 154, 182–192.
<https://doi.org/10.1016/j.epsr.2017.08.016>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Validation of a Robust Neural Real-Time Voltage Estimator for Active Distribution Grids on Field Data

Michael Pertl^a, Philip J. Douglass^b, Kai Heussen^a, Koen Kok^{a,c,*}

^a*Center for Electric Power and Energy, Technical University of Denmark,
Frederiksborgvej 399, Building 776, 4000 Roskilde, Denmark*

^b*Danish Energy Association,*

Vodroffsvej 59, 1900 Frederiksberg C, Denmark

^c*Netherlands Organization for Applied Research, TNO,
Anna van Buurenplein 1, 2595DA The Hague, The Netherlands*

Abstract

The installation of measurements in distribution grids enables the development of data driven methods for the power system. However, these methods have to be validated in order to understand the limitations and capabilities for their use. This paper presents a systematic validation of a neural network approach for voltage estimation in active distribution grids by means of measured data from two feeders of a real low voltage distribution grid. The approach enables a real-time voltage estimation at locations in the distribution grid, where otherwise only non-real-time measurements are available. The method shows robust behavior in all analyzed aspects, which is vital for real world applications. A methodology to select the most relevant input variables and find the best achievable performance for a particular number of inputs is presented. Moreover, the paper shows that the performance is not sensitive to the number of neurons in the hidden layer of the neural network as long as the model is not underdetermined. The paper examines the quantity of historical data needed to establish an adequately functioning model. To accommodate grid evolution and seasonal effects, the impact of different retraining intervals is investigated. Furthermore, the performance of the model during periods of high PV generation is evaluated. The validation shows that accurate voltage estimation models for distribution grids with high share of dispersed generation can be established with approximately one month of historical data. The model has to be retrained every 10 to 20 days to retain estimation mean squared errors below 0.35 V^2 . It was also found that the performance does not decline during times of high PV generation.

Keywords: Active Distribution Grids, Data Driven Methods, Distributed Generation, Neural Networks, Real-Time Voltage Estimation, Renewable Energy Sources

*Corresponding author

Email address: `koen.kok@tno.nl` (Koen Kok)

1. Introduction

Renewable energy sources (RES) are continuously being installed at all voltage levels in today's electric power systems [1]. A large share is being installed in the distribution grid, even at the lowest voltage level. Distribution feeders are transitioning to host both, energy users and producers and, thus, the power flow of distribution grids changes significantly in the presence of dispersed generation units [2]. Moreover, voltage becomes more volatile with the fluctuating power output of photovoltaics (PV) and increasing number of single-phase charging electric vehicles [3, 4]. The operation of distribution grids becomes more challenging as distribution grids transform from traditionally passive behavior to more active behavior with a considerable share of generation. To operate an active distribution system, operators need to increase the observability of distribution grids. Today, observability of distribution systems is generally low due to their large size. Observability usually translates into a need for additional measurement sensors, such as smart meters. Obviously, costs prohibit achieving full observability of distribution grids and, hence, complementary methods must be used. Data driven methods benefit from the availability of offline measurements from different sources and offer a cost-efficient alternative to the installation of additional real-time measurements.

Conventional state estimation approaches, such as [5–7] assume that the network topology and accurate line parameters are given. The state estimation accuracy depends highly on accurate line parameters and topology knowledge and seriously degrades in presence of inaccurate parameters [8, 9].

Unlike conventional state estimation approaches, in the approach described in this paper there is no need to model the network as admittance matrix and no iterative process is needed for the estimation after the model has been established. A major difference in the application point of view, is also that only variables for which prior measurements are available can be estimated. For this work, it is assumed that the network topology remains constant between training and observation. Similar to conventional state estimation, topology changes have to be detected in parallel and accounted for. This high flexibility paired with higher speed, accuracy and efficiency compared to their conventional counterparts makes data driven approaches interesting for complex problems and development of online applications [10].

A hierarchical bottom-up distribution system monitoring approach using neural networks (NNs) was proposed in [11]. This hierarchical approach in [11] splits up the monitoring problem to each voltage level. Local estimators are trained to estimate the voltage at certain nodes at the lowest voltage level by using voltage and current measurements at the

medium-voltage/low-voltage transformer. The estimation results are communicated to the upper-level estimator and, thereby, generating an overall picture of the distribution system.

The authors of [12, 13] study how data from phasor measurement units (PMUs) impact the accuracy of NN-based estimation of voltage magnitude and angle. They conclude that the NN-based estimator including input data from PMUs achieves similar results as a classic state estimation algorithm. The current work is based on less expensive non-synchronized measurements.

A NN with two hidden layers and entropy-based selection of input variables is proposed in [14], and it was found that the selection of appropriate input variables is of crucial importance.

The authors of [15] employ a NN voltage estimator to calculate the voltage profile along a feeder. Remote terminal units (RTUs) send the resulting voltage profile to a master controller aiming at enhancing the operation of an on-load tap-changer (OLTC) transformer for voltage regulation.

In the above works and throughout the literature, NN-based voltage estimation approaches are tested and validated by means of simulation models alone. Generally, a large number of different load flow scenarios are simulated and the results of the simulations are divided into training and test set. In contrast, this work focuses on necessary steps towards implementation in real environment by setting the framework for a neural real-time voltage estimator and validation based on actual distribution grid measurements.

This paper builds on the approach proposed in [16] where the conceptual framework and a numerical implementation for a distribution grid model with three feeders including PV generation was implemented and analyzed. It is proposed to estimate the voltage at specific low voltage (LV) buses by use of NNs trained on voltage and power measurements from substation level only. Various generation and consumption scenarios including reverse power flow scenarios are analyzed in terms of estimation accuracy. The results showed the method to be promising for all analyzed scenarios concluding in the need for a validation in real world environment. This manuscript specifically focuses on the validation of the NN-based voltage estimation approach on field data from a real distribution grid, in particular, estimating the phase-neutral voltage magnitudes (U_a , U_b , U_c) at a downstream bus of a distribution feeder based on available measurements from the substation. A general sketch of an active LV distribution grid which includes distributed generation (DG) among loads is shown in Fig. 1. The substation and the downstream measurement are highlighted in red and blue, respectively. In the training phase, historical measurements from the substation and from the downstream bus are used to train the estimator. In the estimation phase, substation measurements are fed into the estimator and the voltages of the downstream bus

are calculated in real-time (order of milliseconds on a standard laptop). Voltage angles are not considered because they are typically small in LV grids while the voltage magnitude is from paramount interest [17]. This paper presents the intermediate step towards implementation of a neural real-time voltage estimator (NRTVE) in an operating environment, such as integration into an existing SCADA system. A considered application of the proposed approach is real-time voltage estimation at buses where measurements, such as smart meters, are installed, but data is not available in real-time. For these buses, a real-time estimator could be established to increase the observability of the distribution grid. The provided assessment of accuracy and sensitivity will serve online monitoring well. However, reactive applications such as voltage control would impose further engineering requirements and development steps to be considered.

The key contributions of the manuscript are twofold:

- I The framework to establish a highly accurate neural real-time voltage estimator is described.
- II The capabilities and limitations of the approach under practical considerations are analyzed, in particular:
 - i Methodology to select the most relevant input variables and find the best achievable performance for a particular number of inputs.
 - ii It is shown that the performance is not sensitive to the number of neurons as long as the model is not underdetermined.
 - iii The quantity of historical data needed to train an adequately functioning model is analyzed.
 - iv The impact of the retraining interval on the performance of the model is determined.
 - v It is shown that the performance of the model is not sensitive to the level of PV generation.

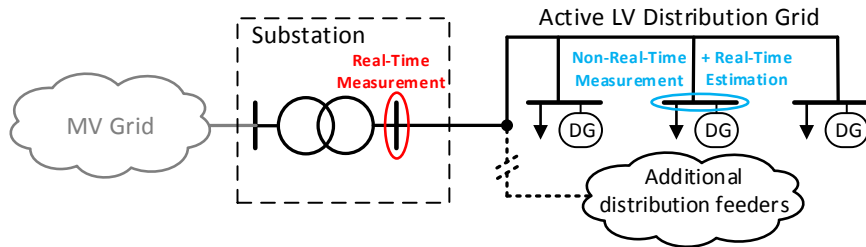


Figure 1: Sketch of an active low voltage distribution grid indicating the substation and downstream measurements

2. Architecture and Training of the Neural Real-time Voltage Estimator

Two different phases of the NRTVE are distinguished: training and real-time estimation, as shown in Fig. 2. The NRTVE is established in the training process by use of an suitable training algorithm. Prior to the training, the architecture and number of neurons in the hidden layer have to be defined. After the training process, the NRTVE can be used for real-time voltage estimation at the specific bus. Distribution network operation is characterized by faults, topology changes and outages. The proposed model is exclusively established for normal operating conditions found in the available data. For estimation under abnormal conditions, a separate model would need to be trained and a change detection would have to be implemented as the characteristics of the disturbed grid are different than in normal operation. Moreover, for changing topologies separate models need to be trained. Here, no topology change occurred.

The calculation of a bus voltage with a trained NN is in the order of milliseconds as it can be directly calculated and no further iterations are needed after the training. That computation time is deterministic is crucial for real-time applications, since it has to be accounted for the worst case. All available input and the three output variables are indicated in Fig. 2. The colors used for the input and output arrows are aligned with the colors in Fig. 1 and 3.

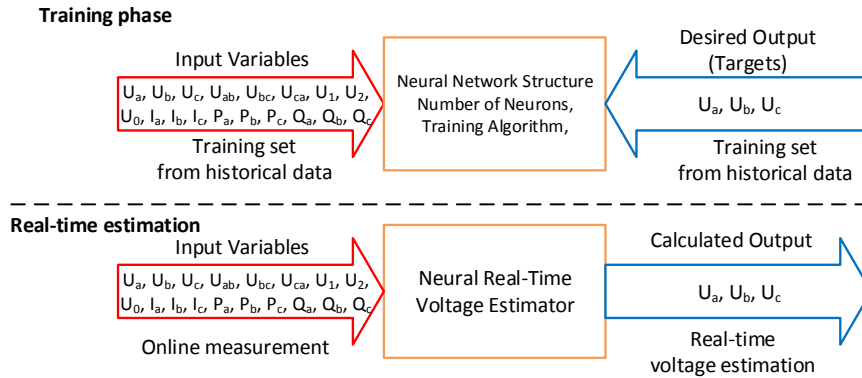


Figure 2: Training phase and real-time estimation phase

2.1. Architecture

A great number of NN architectures can be imagined [18–22]. As this is a fitting problem, a feed-forward NN with one hidden layer is used as that is sufficient for most fitting problems [23]. A multilayer perceptron with hyperbolic tangent-sigmoid neurons in the hidden layer and linear neurons in the output layer is chosen. Additional hidden layers can be added if

the performance is not satisfactory. The term performance refers to the mean squared error (MSE) between the model output and the real measured values. Inputs and outputs are scaled to values between ± 1 since the hyperbolic tangent-sigmoid function in the hidden layer operates between ± 1 . Large inputs would mainly generate values in the saturated area of the sigmoid function and in order to exploit the full flexibility of the transfer function it is advised to scale the inputs to values in the linear region.

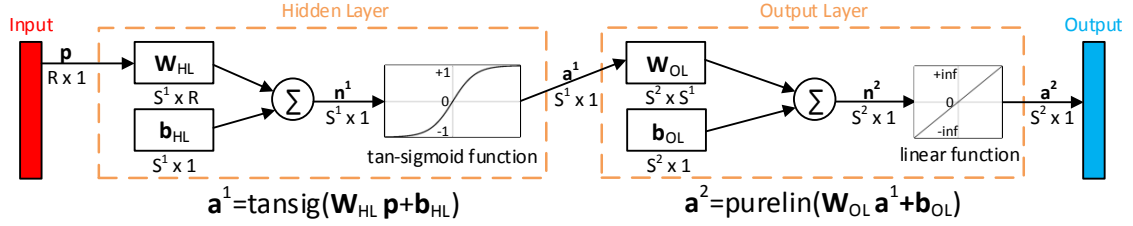


Figure 3: Architecture of the feed-forward neural network. Bold capital letters represent matrices and bold lower case letters vectors. The size of each element is given underneath.

$\mathbf{W}_{HL}/\mathbf{W}_{OL}$ = weight matrices, $\mathbf{b}_{HL}/\mathbf{b}_{OL}$ = bias vectors,
 $S^1 = \# \text{ neurons}$, $R = \# \text{ input variables}$, $S^2 = \# \text{ output variables}$.

2.2. Training Algorithm

The choice of training algorithm depends on the problem type and complexity of the NN. The well known Levenberg-Marquardt algorithm, first introduced in [24], is used to train the NNs because it is usually the fastest training method for function approximation up to a few hundred weights and biases. It is very accurate and it shows superior convergence behavior over other algorithms [23, 25]. The weights and biases are updated according to (1) with \mathbf{J} being the Jacobian matrix, \mathbf{I} the identity matrix, \mathbf{e} the network errors and μ the parameter determining the algorithm behavior during the training.

$$\mathbf{w}_{n+1} = \mathbf{w}_n - [\mathbf{J}^T \mathbf{J} + \mu \mathbf{I}]^{-1} \mathbf{J}^T \mathbf{e} \quad (1)$$

If μ is zero it is Gauss-Newton method and when μ is large it becomes gradient decent. The algorithm starts with a small value μ_{init} . If one step does not yield a smaller error (MSE), μ is increased by multiplication with μ_{inc} . If the the next step produces a smaller error, μ is multiplied by μ_{dec} to approach Gauss-Newton method which converges faster. The work was carried out in the Matlab environment. The training parameters and its corresponding values are shown in Table 1.

The training parameters are set up such that the algorithm stops when either the maximum number of consecutive failed validations is reached, the performance gradient falls below the minimum threshold or the maximum number of training epochs is reached. The

maximum number of epochs and failed validations has been increased above the default values in Matlab’s NN implementation, while the other parameters are suggested default values.

The number of training epochs must be reasonably high in order not to stop training prematurely. The number of maximum failed validations is usually the stopping condition met first which highly speeds up the training and, moreover, prevents overfitting. Validation checks are carried out on the validation data set after every epoch during the training process. However, the validation data set is only used to check the performance; it is not used for the training, i.e. only the training set is used for training.

The weights and biases are initialized using the technique from Nguyen-Widrow [26] because it highly improves the speed of training and the optimality of the achieved solution [23, 25]. As there is a random factor in the initialization, different results are obtained by different trainings. Therefore, every training is rerun five times and the best solution is selected to get a solution which approaches the global minimum, as suggested in [27].

3. Available data for the analysis

This section describes the data that was available to conduct the analysis. All measurements were taken in a LV distribution grid. Seven power quality meters, which comply with measurement class A defined in IEC 61000-4-30 [28], are placed in two different feeders. Both feeders include loads and rooftop PV generation. Two meters are placed at the substation and five downstream the feeder. Considering Fig. 1, two meters are located at the red measurement point and five at the blue one, respectively. The distribution feeders are located in a residential area of Copenhagen region. For reasons of confidentiality, exact specifications cannot be disclosed.

Table 1: Training parameters of the Levenberg-Marquardt algorithm

Parameter abbreviation	Value	Explanation
<i>epochs</i>	2500	max. epochs to train
<i>goal</i>	0	performance goal (MSE)
<i>min_grad</i>	10^{-7}	minimum improvement from one epoch to the next
<i>val_fail</i>	10	maximum consecutive validation fails
μ_{init}	10^{-3}	initial mu
μ_{dec}	0.1	mu decrease factor
μ_{inc}	10	mu increase factor
μ_{max}	10^{10}	maximum mu

The measurements were taken over a period of approximately 1.5 years starting from 01.10.2014 until 14.05.2016. The resolution is a mixture of 1-min and 10-min measurements, i.e. in some periods a measurement is averaged over 10 minutes and in other periods averaged over one minute. As synchronized data from exactly two different meters is needed, the measurement data has to be filtered to extract the measurements with the same time stamp of the considered meters. The five possible combinations of substation and downstream meters including their set size, measurement period and PV generation are shown in Table 2. All datasets include the available measurements shown in Table 3.

Table 2: Available meter combinations including set size, measurement period and indication of PV generation on the feeder. $S = \text{substation meter}$, $D = \text{downstream meter}$

Meter combination	$S_1 - D_1$	$S_1 - D_2$	$S_1 - D_3$	$S_2 - D_4$	$S_2 - D_5$
Data points	274,739	274,515	213,660	83,985	84,693
Period	01.10.2014-14.05.2016	01.10.2014-14.05.2016	30.11.2015-14.05.2016	01.10.2014-14.05.2016	01.10.2014-14.05.2016
PV on feeder	yes	yes	yes	yes	yes

Table 3: Measured variables

Variable	Explanation
U_a, U_b, U_c	phase-to-neutral voltages (V)
U_{ab}, U_{bc}, U_{ca}	phase-to-phase voltages (V)
U_1, U_2, U_0	positive-, negative- and zero-sequence voltage (V)
I_a, I_b, I_c	currents (A)
P_a, P_b, P_c	active powers (W)
Q_a, Q_b, Q_c	reactive powers (VAr)

4. Selection of input variables

The performance of the trained NN depends on the selected input variables. A trade-off between the number of selected inputs and the model performance must be found. Input variables with negligible or low impact on the output variables should be excluded to reduce the model complexity. As input variables are excluded, the performance of the NN will get worse because even low impact input variables contain a certain amount of useful information which is lost when the variable is not used. To identify the most relevant input variables, relationships between the input and output variables, which can be linear but also non-linear, must be discovered. There is no general algorithm to optimally select the input variables, but the most commonly used relevance measure found in the literature is the

185 Pearson correlation [29]. Accordingly, the input variables are ranked by the order of their correlation with the output variables according to (2), where X and Y are the two variables and k is the number of data points of each variable.

$$R_{(X,Y)} = \frac{\sum_{i=1}^k (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^k (x_i - \bar{x})^2 \sum_{i=1}^k (y_i - \bar{y})^2}} \quad (2)$$

However, if the input variables are correlated themselves, the correlation ranking approach does not consider this redundancy. Hence, it is not suitable to find a trade-off between
 190 selected input variables and performance. In the following, the ranking approach is elaborated by use of a dataset which contains measurements of the meter combination $S1 - D_1$ from 01.10 – 01.11.2014, corresponding to 3124 data points for training and 670 for validation and testing each. In order to produce comparable results, the same data sets are used for training, validation and testing throughout the whole input selection process. The
 195 number of neurons in the hidden layer is set to 10 and is not changed during the input selection process. The exact number of neurons is not of great importance during the selection process because validation checks are carried out during the training process. However, the number of chosen neurons must be high enough so that the model is not underdetermined.

4.1. Ranking approach

200 The correlation between the input and output variables and among the inputs are shown in Fig. 4 and 5, respectively. Note, here the estimated output is used to calculate the correlation, i.e. a NN is trained, then the correlation is calculated with the inputs and estimated outputs of the test set. Some of the input variables are highly correlated with the output variables and, as expected, some of the inputs are highly correlated with each other.
 205 The results of the correlation assessment can be viewed from an electrotechnical perspective.

Input/Output correlation: The phase-neutral, phase-to-phase and positive-sequence voltages are highly positively correlated with the outputs, e.g. if the voltage at the substation is low, the voltage at buses at buses downstream the feeder will also be low and vice versa. On the other hand, the currents and powers have a negative correlation to the outputs, e.g.
 210 if the current (power) flow at the substation increases, the voltage drop over the line will increase and the voltage at buses downstream the feeder will be lower. The mentioned relationships start to deviate as soon as DG is installed at the feeder, e.g. higher bus voltages when close to producing PV units.

Input/Input correlation: Some of the input variables are highly correlated with each
 215 other. The correlation among the phase-neutral and phase-to-phase voltages is very high despite the unbalanced conditions at the feeder. Naturally, the positive-sequence voltage

is highly correlated with the phase-neutral and phase-to-phase voltages. The currents and active powers are also highly correlated due to their electrical interrelation. However, slight negative correlations between currents (active powers) and the voltages are present. Moreover, slight positive correlations between reactive and active powers are present.

The correlation among inputs suggests that the input variables contain redundant information which has to be considered in the input selection process. Hence, a ranking approach, which considers the correlation between inputs and outputs and among the inputs, is proposed to find an optimal trade-off between number of selected inputs and performance of the model. Two important preceding considerations for the ranking approach are made:

- As the correlation between any input variable and the three output variables is virtually the same, the average value is used, i.e. when referring to the correlation of an input variable to the output, the average to all three output variables is meant.
- The input variables are grouped into six categories. The categories (indicated in curly brackets) are chosen from an electrotechnical perspective because an established NN should either include all or none of the associated group variables. Otherwise the input selection will consist of a mix of node and branch parameters. Hence, it was found suitable to categorize the variables accordingly: 1) phase-neutral voltages $\{U_{ph}\}$, 2) phase-to-phase voltages $\{U_{pp}\}$, 3) voltage's sequence-components $\{U_{0,1,2}\}$, 4) currents $\{I\}$, 5) active powers $\{P\}$, 6) reactive powers $\{Q\}$.

Note: Here, the variables are grouped into categories during the input selection process from an electrotechnical perspective, but if these considerations do not apply, the method can also be applied to individual variables.

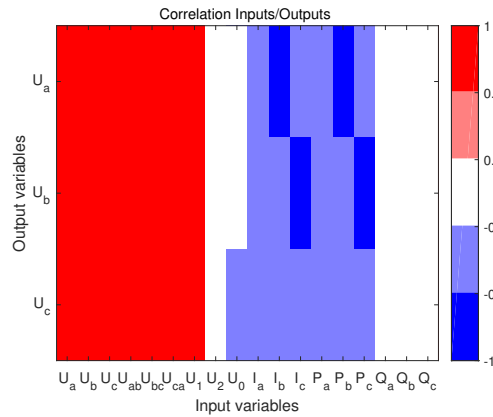


Figure 4: Correlation between input and output variables

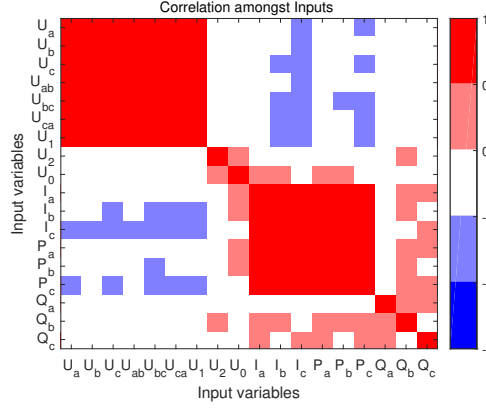


Figure 5: Correlation among input variables

240 The variable importance VI_C for a particular input category C is introduced as a ranking index. The calculation of VI is shown in (3) and it consists of two terms. The first term $R_{I,O}$ describes the average correlation of the particular input category C (which includes several variables X) with the output Y whereas Ind_C denotes the indices of the variables within a category. Note that it is the absolute value of the correlation which is averaged,
 245 with values close to 1 indicating the strongest correlation. The second term $R_{I,I}$ describes the average correlation between the particular input category and the other input variables. This value does not necessarily need to be close to zero, but it has to be optimized if the model complexity is to be reduced while maintaining acceptable performance. In order to express the impact of redundancy, an additional parameter $\beta \geq 0$ is introduced. If $\beta = 0$,
 250 the variable importance equals the correlation between inputs and outputs, ignoring the correlations between input categories. The greater the value of β , the more importance is given to reducing the redundancy of input variables. The choice of β depends on the purpose of the developed NN, e.g. β can be chosen to be zero if the model complexity does not matter, but if the aim is to sort out as many input variables as possible while keeping
 255 satisfying performance, β must be increased. To find an optimal trade-off between model complexity and performance β must be varied.

$$VI_C = R_{I,O} - \beta \cdot R_{I,I} \quad (3)$$

with

$$R_{I,O} = \frac{1}{n_C} \sum_{\substack{i=1 \\ i \in Ind_C}}^{n_C} |R_{(X_i,Y)}| \quad \text{and} \quad R_{I,I} = \frac{1}{n_C} \sum_{\substack{i=1 \\ i \in Ind_C}}^{n_C} \frac{1}{m - n_C} \sum_{\substack{j=1 \\ j \notin Ind_C}}^m |R_{(X_i,X_j)}| \quad (4)$$

where n_C is the number of variables in the category and m the total number of variables.

A backward elimination approach is used to sort out the least important variable categories step-by-step, i.e. all variables are selected initially and one after another category with the lowest ranking gets eliminated. The approach is computationally heavier than other approaches because a NN has to be trained for all available input variables and with varying β -values, however, this computation only needs to be done once, offline. The advantage of the backward elimination approach is that it shows the maximum achievable performance when all input variables are used and how it declines when some of them are sorted out. However, it is a 'greedy' algorithm because it eliminates the least relevant variables one by one while the global optimum might be found only by trying all possible combinations. As this is computationally infeasible, a greedy approach is used instead. After the training of a NN with all input variables, the variable to be eliminated has to be determined according to (3). In order to achieve the different input variable combinations, the elimination process needs to be repeated with different β -values. Beta is varied between 0 and 3 in 0.2 increments. The range of the variation originates from the fact that the largest correlation among input categories is about one third of the input-output correlation.

The results of the VI calculation for three selected β -values is shown in Fig. 6. The category with the lowest VI index is eliminated. It is clearly visible from Fig. 6 that the results for different β -values deviate significantly from each other. When β equals 0 and 1, the reactive powers have the lowest VI index. When β equals 3, the currents are eliminated first. This elimination procedure is carried out until one only variable category is left.

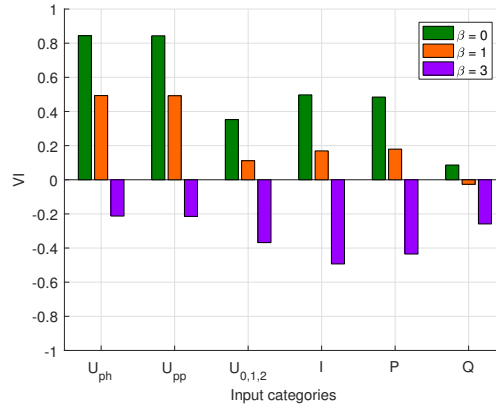


Figure 6: Variable importance for different values of β when all input categories are selected.

The results of the elimination process are shown in Fig. 7. The left plot shows the results for all beta variations and the right plot for three selected beta values. NNs with varying input variable combinations achieve different performances. The black line shows the best

achievable performance for a given number of input variables. The performance declines slightly when the first two categories are sorted out, i.e. 12 input variables are selected. It starts to decline more between 12 and six variables when the next two categories are eliminated and it highly drops when one category is left only. Considering the blue lines, it can be seen that the variation of beta is necessary in order to find a trade-off solution because the results change significantly with variation of beta and no one value of beta gave the smallest error for all number of input variables.

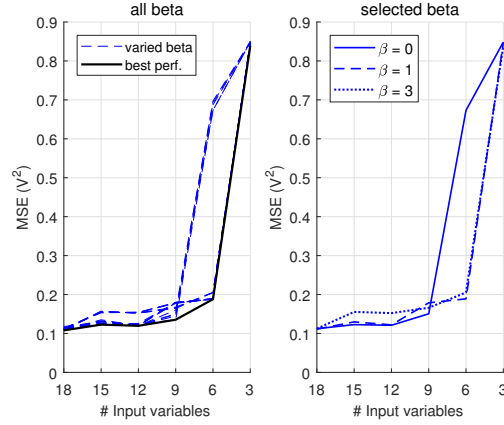


Figure 7: The blue lines represent the performance throughout the input elimination process with β varied between 0 – 3.

In order to discuss the elimination process more in detail, three 'interesting' β -values are shown separately in the right plot of Fig. 7. The stepwise eliminated categories for these three cases are shown in Table 4. Comparing the performance of the first two cases ($\beta = \{0, 1\}$), it can be seen that the performance between 18 and 12 inputs is the same in both cases because the same categories were sorted out, namely Q and $U_{0,1,2}$. In the next step two different categories are sorted out. In one case it is the active powers and in the other the currents. Recalling from Fig. 5, the active powers and currents are highly correlated and, thus, the performance with nine inputs shows only slight differences. However, the next step shows the most interesting difference between the two cases. When β equals zero, the currents are sorted out next, which means that no branch variables are included in the inputs anymore, but node variables only, i.e. phase-neutral and phase-to-phase voltages. In the other case, the phase-to-phase voltages are sorted out and the input variables are composed of node and branch variables, i.e. phase-neutral voltages and active powers. In the third case ($\beta = 3$), the variable selection differs completely from the prior cases. It can be seen that the error increases more at the beginning when the first two categories are sorted out due to the fact that the correlation among input variables is given a high weight in the selection process.

That highlights the importance of considering the correlation among inputs and between
 305 inputs and outputs to find the optimal combination of parameters, which lies somewhere in
 between and is found by varying the parameter beta.

Table 4: Eliminated input variables for the three selected β -values

# inputs		18	15	12	9	6	3 (final)
eliminated category	$\beta = 0$	Q	$U_{0,1,2}$	P	I	U_{pp}	U_{ph}
	$\beta = 1$	Q	$U_{0,1,2}$	I	U_{pp}	P	U_{ph}
	$\beta = 3$	I	$U_{0,1,2}$	U_{ph}	Q	P	U_{pp}

The assessment was also carried out for the other meter combinations using datasets from
 different periods of the year. The results showed the same behavior, but with different input
 parameter combinations. Therefore, the input selection assessment must be carried out for
 310 each feeder separately because every feeder has its particular electrical characteristics.

Concerning the sensitivity of results to the number of neurons in the NN, Fig. 8 shows the
 test performance of the NN with the best performance input selection for different number
 of neurons in the hidden layer. It can be seen that the test performance stabilizes in all
 cases before 10 neurons are reached and, thus, the choice to use 10 neurons during the
 315 selection process is reasonable. Twelve input variables are used for the further studies. The
 four categories U_{ph} , U_{pp} , P and I are chosen. The results showed that the reactive powers
 and sequence components are sorted out when achieving best performance with 12 input
 variables. Moreover, the input selection process showed clearly that satisfying performance
 is maintained when these two categories are eliminated.

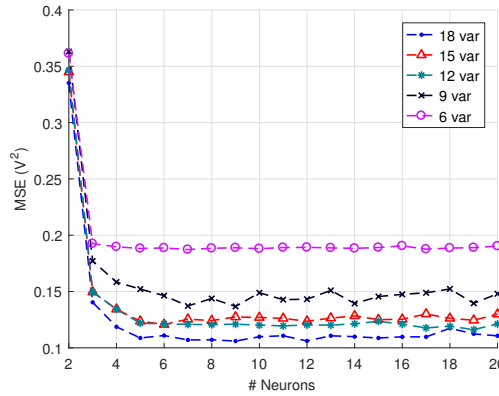


Figure 8: Performance of NNs with the determined input selections for different number of neurons

320 5. Data quantity and retraining analysis

This section analyses the relationship between the performance of NN models and the historical data quantity that is used to establish them and how often they have to be retrained to be suitable for real-time application. The section aims at answering two questions:

- i How much historical data is needed to create a model with a satisfying performance for
325 real-time application?
- ii When does the model have to be retrained in order to maintain satisfying performance for real-time application?

*Note: Satisfying performance is achieved when the accuracy of the estimation is comparable to the measurement uncertainties of meters, such as meters of class A and B, defined
330 in IEC 61000-4-30, which shall not exceed ± 0.1 and $\pm 0.5\%$ of the declared input voltage, respectively.*

5.1. How much historical training data is needed?

The analysis is carried out for four meter combinations in order to derive meaningful information about the needed data quantity. In order to consider seasonal effects, one full
335 year of measurements was used from each meter combination. Generally speaking, a point in time is chosen to represent the present which is denoted by $t_{0,n}$ with $n \in [1, 73]$, i.e. the analysis is carried out for 73 different t_0 from 01.01.2015 to 27.12.2015 in 5 day steps. Time before and after t_0 represents the past and future, respectively. Fig. 9 shows a graphical representation of the approach. The initial condition is depicted in black letters, whereas
340 $t_{0,1}$ denotes the starting point of the analysis. The second out of the 73 iterations is shown in green color. The final iteration at the end of the year is shown in purple. The different steps of the calculation procedure are explained in the following and are referred to Fig. 9:

- I Set starting point t_0 , i.e. $t_{0,1}$ which is in this case 01.01.2015 00:00.
- II Train a NN with data from the past 3 days. The training and validation dataset is
345 denoted by $TVS_{1,1}$. The historical data is randomly split into 85 % for training and 15 % for validation.
- III Evaluate the performance of the established model on the test set denoted by TS_1 which consists of the 7 days following $t_{0,1}$.
- IV Repeat step 2 and 3, increasing the amount of data that is used for training and
350 validation by 3 days in each step until the maximum considered amount of 90 days is reached, i.e. $[TVS_{1,1} \rightarrow TVS_{2,1} \dots TVS_{30,1}]$.

V Once step 4 is completed, i.e. the performance of the NN with training set size from 3 to 90 days is evaluated, t_0 is shifted forward in time by 5 days (e.g. $t_{0,1} \rightarrow t_{0,2} \dots t_{0,73}$) and steps 2-4 are repeated.

355 VI The procedure finishes when the end of the year (27.12.2015) is reached.

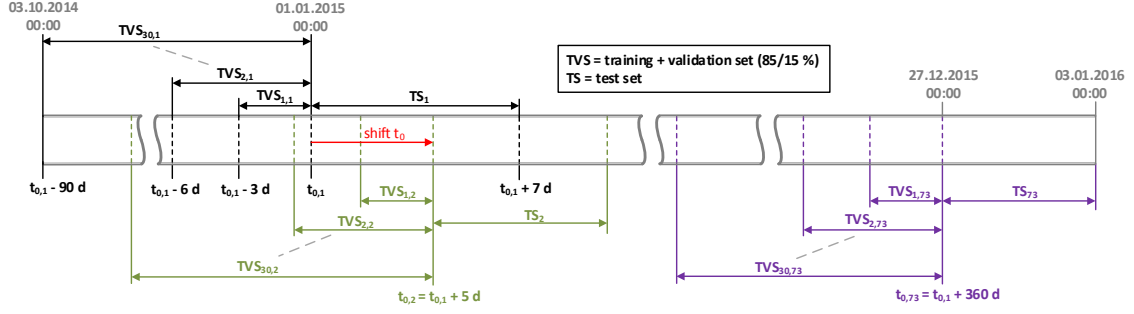


Figure 9: Visualization of the approach for the data quantity analysis. Initial condition = black, second iteration = green and last iteration = purple.

In order to clearly visualize the results, they are averaged per month as shown in Fig. 10. The results of the four different meter combinations show similar behavior. It must be pointed out that the general statement for machine learning that more data will provide better performance is not valid in this case. It can be seen clearly that the performance stabilizes at about 30 days of data and there is even a slight tendency that performance starts to decline afterwards. The analysis is carried out as close as possible to real world conditions by taking the test set data from the "future", which would be the same when applied in practice. A possible explanation for the stabilization of performance could be that the data which is too far in the past does not reflect the current grid situation and, therefore, the more recent data is sufficient to establish a suitable model. It can be seen that the performance of the approach differs for the different meter combinations. The best performance is achieved for the meter combination $S_2 - D_5$, whereas the worst for $S_2 - D_4$. At the combinations $S_1 - D_1$ and $S_1 - D_2$ there is a tendency that the performance is slightly worse during summer months than during the rest of the year which may be inferred from the larger amount of installed PV generation compared to the other feeder.

5.2. How often does the model have to be retrained?

According to the outcome of the data quantity analysis, the performance of the voltage estimation model stabilizes at around 30 days of historical data. Hence, 30 days of historical data are considered in the training process of the further analysis. Fig. 11 shows the visualization of the retrain assessment procedure. Similar as in the data quantity analysis, one

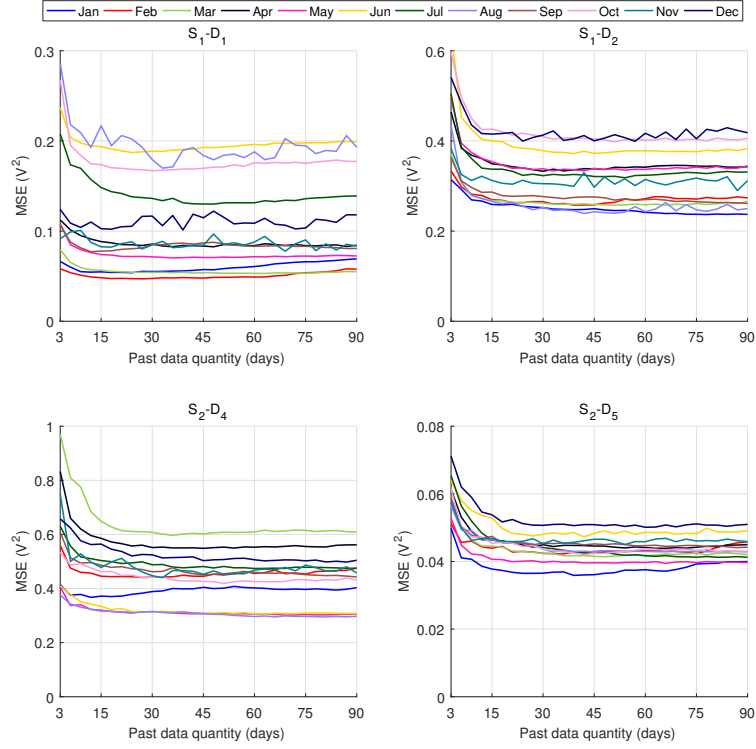


Figure 10: Results of the data quantity analysis for four different meter combinations

full year of data from 01.01.2015 until the 27.12.2015 is taken into account. For consistency, the same notation is used in Fig. 11 as in Fig. 9. The steps of the procedure are explained in the following:

I Set starting point t_0 , i.e. $t_{0,1}$ which is again 01.01.2015 00:00.

II Train a NN with data from the past 30 days. The training and validation dataset is denoted by $TVS_{10,1}$. The indices 10 originates from the fact that a granularity of 3 days was used before, i.e. $3 \text{ days} \cdot 10 = 30 \text{ days}$. The data is randomly split into 85 % for training and 15 % for validation.

III The error of the estimation from the model is calculated for the test set denoted by TS_1 which set size depends on the retrain interval RI, e.g. it is 7 days when the model is retrained every 7 days.

IV Shift t_0 forward in time by the amount of days which are defined by the retrain interval RI (e.g. $t_{0,1} \rightarrow t_{0,2}$) and repeat steps 2 and 3 with respective TVS and TS .

V Once the errors are calculated for the whole year, the performance of the model is evaluated over the whole period, i.e. MSE is calculated.

VI Repeat step 1 to 5 with different RIs.

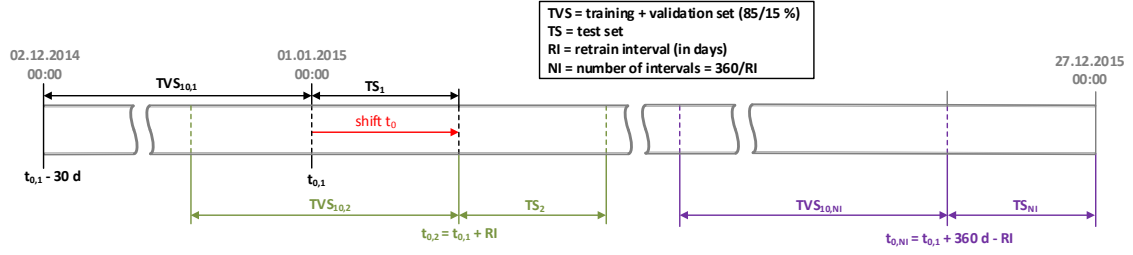


Figure 11: Visualization of the approach for the retrain analysis. Initial condition = black, second iteration = green and last iteration = purple.

The results of the retraining analysis for the same four meter combinations as in the previous section with varying retraining intervals are shown in Fig. 12. The plot shows the results on the y-axis only up to $MSE = 0.8 V^2$ and the retrain interval on an logarithmic x-axis. There is a clear trend that the performance declines with an increased retrain interval. The performance of all meter combinations declines only slightly up to an RI of 10 days. The performance of $S_1 - D_1$ and $S_1 - D_2$ drops significantly when the RI is greater than 40 days. However, the performance for $S_2 - D_4$ and $S_2 - D_5$ is much more stable and does not decline so much. It can be concluded that the RI highly depends on the properties of the feeder, e.g. amount of PV generation. Certainly, the modelling errors can be kept at an reasonably low level with adequate choice of the RI. Since the training procedure only takes a few seconds (on a standard computer with Intel i7 processor with 8 GB RAM) with a training/validation set of 30 days with a resolution of about 1 minute averaged measurements.

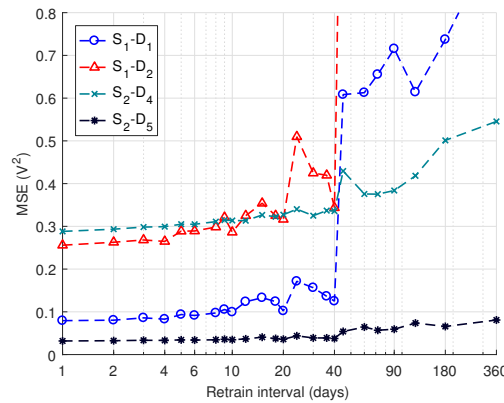


Figure 12: Model performance versus retraining interval

In order to quantify the quality of the estimation, the confidence interval of the expected error of the estimation is calculated. Fig. 13 shows the error distribution of the meter combination $S_1 - D_1$ for a retrain interval of 1 day over the evaluation period of one full year. It can be clearly seen that the errors are normally distributed. Hence, considerations of

the normal distribution can be applied. Therefore, the mean μ and the standard deviation σ can be calculated and they are equal to 7.8 mV and 333.7 mV , respectively. Fig. 13 shows the mean and 3σ - confidence interval of the estimation. According to the normal distribution theory, 99.73 % of the estimations lie between an interval of $\mu \pm 3\sigma$ which is equal to $7.8 \pm 3 \cdot 333.7\text{ mV} = [-0.9934\text{ V } 1.0090\text{ V}]$.

To evaluate how the retraining interval affects the quality of the estimation, the mean and respective 3σ - confidence interval for the four considered meter combinations versus the retrain interval is shown in Fig. 14. A general trend can be observed for all meter combinations, namely that the mean value remains for all of them close to zero. However, the standard deviation and, therefore, the confidence interval increases with longer retrain intervals. The results also show different confidence intervals which, however, correspond to the results of the performance of the estimation shown in Fig. 12, i.e. the width of the confidence interval reflects the performance of the model. As shown in Fig. 12, the best performance is achieved at the meter combination $S_2 - D_5$ and, indeed, the same meter combination shows the narrowest confidence interval. The same applies for the other meter combinations in descending order.

In order to assess the model performance during high PV in-feed, a scatter plot of the active power flow at the substation meter S_1 versus the estimation error of the meter combinations $S_1 - D_1$ and $S_1 - D_2$ for June 2015 with a retrain interval of 1 day is shown in Fig. 15. These meter combinations are chosen because this feeder contains more PV generation. The active power is presented on the x-axis and the error on the y-axis. Positive active power means 'normal' power flow direction into the feeder and negative means reverse power flow due to excess of PV generation over consumption. The maximum occurring power flow into the feeder is approximately 14 kW and the largest reverse power flow peaks at about -11 kW , i.e. a broad range of power flow conditions is covered during the evaluation

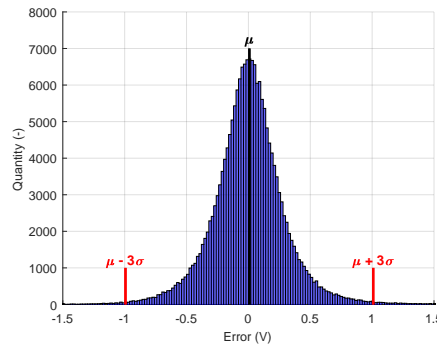


Figure 13: Error distribution of $S_1 - D_1$ for a RI of 1 day with $\mu = 7.8\text{ mV}$, $\sigma = 333.7\text{ mV}$.

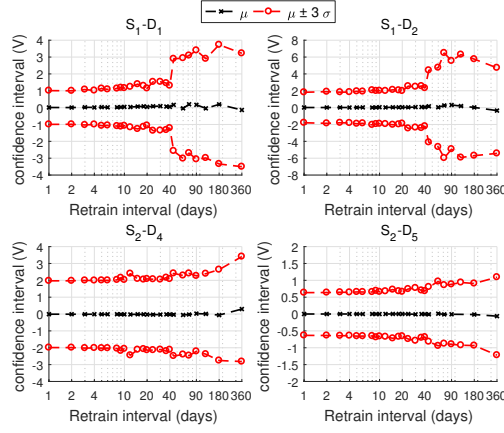


Figure 14: Mean error μ with $3\text{-}\sigma$ confidence interval of the voltage estimations for all meter combinations

period. However, no significant differences in the model performance over varying load flow conditions can be identified, i.e. the model performance does not perform worse during high PV in-feed compared to 'normal' conditions without any distributed generation. That means, that the NN estimation model is suitable for high accuracy voltage estimation under presence of distributed generation.

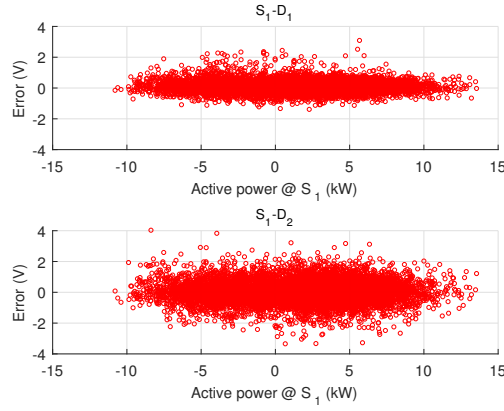


Figure 15: Active power flow at substation versus estimation error of the meter combinations $S_1 - D_1$ and $S_1 - D_2$ for June 2015.

6. Conclusion

This paper presented the validation of a neural real-time voltage estimator for active distribution grids. In particular, the phase-neutral voltages at a bus, downstream from the feeder, are estimated using measurements from the substation of the feeder. The model can be used for real-time voltage estimation as the output of the model can be deterministically

calculated from the input after the model has been established. A possible application of the approach is real-time estimation at buses where measurements, such as smart meters, are installed, but data is not available in real-time. Historical measurement data can be used to establish and retrain the model in appropriate time intervals. Based on data gathered from an operating public LV power system, the method is shown to be robust and, therefore, applicable in real world. The accuracy achieved by the method is in the same order of magnitude as the measurement uncertainty of a physical measurement device.

A novel approach to select the appropriate inputs of the model has been introduced. The correlation between inputs and outputs and among inputs is used to eliminate input variables with low impact on the output error. Reducing redundancy of inputs keeps model complexity, and thus computation time, within reasonable limits. It was shown that the presented input selection approach is capable of sorting out input variables to determine the best achievable performance with a certain amount of input variables. Analysis of the performance as a function of the quantity of historical data has clearly showed that the general statement that better models are produced with more data is not valid in this case. Instead, the assessment showed that the performance of the model stabilizes when about one month of historical data is used in the training process and does not improve further if more data is used. There is even a slight trend that the model performance declines when more data is used.

The analysis of the impact of the retraining interval on the model performance showed that, generally, the performance is better with shorter retraining intervals. It was observed that the performance declines only slightly up to a retrain interval of 10 days. However, the performance for some meter combinations drops significantly for larger retrain intervals while it only declines slightly for others. Therefore, it is concluded that the retrain interval should be kept below 20 days.

It was shown that the error of the estimation is normally distributed and, thus, the confidence interval of the estimation error was calculated over different retrain intervals. Corresponding to the results of the performance analysis over different retrain intervals, the standard deviation of the estimation error increases with increasing retrain intervals. The width of the 3σ - confidence interval which includes 99.73 % of the calculated estimations is different for every meter combination, but at maximum $3\sigma \approx \pm 2$ V. That means, that the produced error, for retrain intervals under 20 days, is approximately 0.87 % of the nominal phase-neutral voltage (230 V), which is still acceptable considering the measurement uncertainty of 0.5 % for class B meters defined in IEC 61000-4-30. Besides, the method is also found to be accurate during periods of high PV generation, as the errors during periods of high load, and periods of high PV production do not show noticeable differences.

Further research on the robustness of the estimation to wrong or missing data has to
 480 be conducted, as corrupted data certainly affects the performance of the model. Moreover,
 the approach could be extended to include additional input variables, e.g. solar irradiation,
 aiming at improving the performance of the estimation.

Acknowledgment

Michael Pertl is a PhD student at the Technical University of Denmark (DTU) and is
 485 supported by the EU FP7 project ELECTRA (grant: 609687) and the Danish Research
 Project ELECTRA Top-up (grant: 3594756936313). More information can be found at
electrairp.eu. Koen Kok's contribution was financed by the TNO Early Research Program
 on Energy Storage & Conversion (ERP EC&S) through the SOSENS project.

References

- 490 [1] KEMA Consulting, Integration of Renewable Energy in Europe, Tech. rep., Imperial Collaage London (2014).
- [2] T. J. Hammons, Integrating renewable energy sources into European grids, *International Journal of Electrical Power and Energy Systems* 30 (8) (2008) 462–475. doi:10.1016/j.ijepes.2008.04.010.
- [3] K. Knezovic, M. Marinelli, R. J. Moller, P. B. Andersen, C. Træholt, F. Sossan, Analysis of voltage
 495 support by electric vehicles and photovoltaic in a real Danish low voltage network, in: *Universities Power Engineering Conference (UPEC)*, Cluj-Napoca, 2014, p. 6. doi:10.1109/UPEC.2014.6934759.
- [4] K. Knezović, M. Marinelli, Phase-wise enhanced voltage support from electric vehicles in a Danish low-voltage distribution grid, *Electric Power Systems Research* 140 (2016) 274–283. doi:10.1016/j.epsr.2016.06.015.
- 500 [5] S. Sarri, M. Paolone, R. Cherkaoui, A. Borghetti, F. Napolitano, C. A. Nucci, State estimation of Active Distribution Networks: Comparison between WLS and iterated kalman-filter algorithm integrating PMUs, in: *IEEE PES Innovative Smart Grid Technologies Conference Europe*, Berlin, 2012, pp. 1–8. doi:10.1109/ISGTEurope.2012.6465871.
- [6] J. Wu, Y. He, N. Jenkins, A robust state estimator for medium voltage distribution networks, *IEEE Transactions on Power Systems* 28 (2) (2013) 1008–1016. doi:10.1109/TPWRS.2012.2215927.
 505 URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6313963>
- [7] A. Ranković, B. M. Maksimović, A. T. Sarić, A three-phase state estimation in active distribution networks, *International Journal of Electrical Power & Energy Systems* 54 (2014) 154–162. doi:10.1016/j.ijepes.2013.07.001.
- 510 [8] F. F. Wu, Power system state estimation: a survey, *International Journal of Electrical Power & Energy Systems* 12 (2) (1990) 80–87. doi:10.1016/0142-0615(90)90003-T.
- [9] A. Monticelli, Electric power system state estimation, *Proceedings of the IEEE* 88 (2) (2000) 262–282. doi:10.1109/5.824004.
- [10] L. H. Hassan, M. Moghavvemi, H. a.F. Almurib, O. Steinmayer, Current state of neural networks
 515 applications in power system monitoring and control, *International Journal of Electrical Power & Energy Systems* 51 (2013) 134–144. doi:10.1016/j.ijepes.2013.03.007.
 URL <http://www.sciencedirect.com/science/article/pii/S014206151300104X>
- [11] M. Ferdowsi, A. Lowen, New monitoring approach for distribution systems, in: *IEEE International Conference on Instrumentation and Measurement Technology*, Montevideo, 2014, pp. 1506–1511.
 520 URL http://ieeexplore.ieee.org/xpls/abs/_all.jsp?arnumber=6860997
- [12] O. Ivanov, M. Gavrilas, State Estimation with Neural Networks and PMU Voltage Measurements, in: *International Conference and Exposition on Electrical and Power Engineering (EPE)*, Iasi, 2014, pp. 983–988.

- [13] O. Ivanov, M. Gavrilas, B. Neagu, Intelligent monitoring and control in transmission and distribution networks, in: International Conference on Optimization of Electrical and Electronic Equipment (OPTIM), Bran, 2014, pp. 185–191. doi:10.1109/OPTIM.2014.6850918.
- [14] Y.-Y. Hsu, C.-C. Yang, Fast voltage estimation using an artificial neural network, Electric Power Systems Research 27 (1993) 1–9.
- [15] R. Kamali, R. Sharifi, H. Radmanesh, S. Fathi, Online voltage estimation for distribution networks in presence of distributed generation, Indian Journal of Science and Technology 9 (18) (2016) 1–5. doi:10.17485/ijst/2016/v9i18/71484.
- [16] M. Pertl, K. Heussen, O. Gehrke, M. Rezkalla, Voltage Estimation in Active Distribution Grids Using Neural Networks, in: IEEE Power and Energy Society General Meeting, Boston, 2016, p. 5.
- [17] M. Paolone, A. Borghetti, C. A. Nucci, A synchrophasor estimation algorithm for the monitoring of active distribution networks in steady state and transient conditions, in: 17th Power Systems Computation Conference (PSCC), Stockholm, 2011, pp. 1–8.
- [18] S. Haykin, Neural networks - a comprehensive foundation, 2nd Edition, Pearson Education, 1999. arXiv:arXiv:1312.6199v4, doi:10.1017/S0269888998214044.
- [19] M. Y. Rafiq, G. Bugmann, D. J. Easterbrook, Neural network design for engineering applications, Computers and Structures 79 (2001) 1541–1552. doi:10.1016/S0045-7949(01)00039-6.
- [20] B. M. Wilamowski, Neural Network Architectures and Learning, in: International Conference on Industrial Technology (ICIT), Maribor, 2003, pp. 1–12.
- [21] B. M. Wilamowski, Neural network architectures and learning algorithms, IEEE Industrial Electronics Magazine (2009) 56–63doi:10.1109/ICIT.2011.5754336.
- [22] B. Wilamowski, Neural Network Architectures, in: Intelligent Systems, 2nd Edition, Taylor & Francis, 2011, Ch. 6, pp. 1–17. doi:doi:10.1201/b10604-9. URL <http://dx.doi.org/10.1201/b10604-9>
- [23] M. T. Hagan, H. B. Demuth, M. H. Beale, O. De Jesús, Neural Network Design, 2nd Edition, PWS Publishing Co. Boston, MA, 1996.
- [24] D. W. Marquardt, An Algorithm for Least-Squares Estimation of Nonlinear Parameters, Journal of the Society for Industrial and Applied Mathematics 11 (2) (1963) 431–441. arXiv:arXiv:1011.1669v3, doi:10.1137/0111030.
- [25] M. T. Hagan, M. B. Menhaj, Training Feedforward Networks with the Marquardt Algorithm, IEEE Transactions on Neural Networks 5 (6) (1994) 989–993. doi:10.1109/72.329697.
- [26] D. Nguyen, B. Widrow, Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights, in: International Joint Conference on Neural Networks (IJCNN), San Diego, 1990, pp. 21–26.
- [27] L. Hamm, B. W. Brorsen, M. T. Hagan, Comparison of Stochastic Global Optimization Methods to Estimate Neural Network Weights, Neural Processing Letters 26 (3) (2007) 145–158. doi:10.1007/s11063-007-9048-7.
- [28] IEC, IEC Standard 61000-4-30: Testing and measurement techniques -power quality measurement methods (2016).
- [29] R. May, G. Dandy, H. Maier, Review of Input Variable Selection Methods for Artificial Neural Networks, in: Methodological Advances and Biomedical Applications, InTech, 2011, pp. 19–44.